<u>**AMENDMENTS TO THE SPECIFICATION:**</u>

Please amend the paragraph staring on page 37, line 8 to read as follows:

--Fig. 40 represents an example of processing in which rendering engine 41 executes a Hough transformation. First, in step S191, θ is initialized to 0. This θ represents the rotation angle of the input image, as shown in Fig. 41. In step S192, image data is computed by applying an affine transformation of angle θ to the input image. This angle θ defines the coefficients a to d in equation (11) described above.--

Please amend the paragraph staring on page 37, line 13 to read as follows:

--Next, processing advances to step S193, where bilinear reduction in the y-direction is executed. Details about this bilinear reduction in the y-direction are described later for the flowchart in Fig. 42. This processing causes the input image that had been rotated by the angle θ to be expressed as a single straight line in the y-direction, as shown by a typical example in Fig. 41.--

Please amend the paragraph staring on page 37, line 17 to read as follows:

--Next, processing advances to step S194. In step S194, processing is executed in which the point sequence that had been linearly reduced in step S193 is written at the position corresponding to the angle θ in rendering area 52.--

Please amend the paragraph staring on page 37, line 20 to read as follows:

--Next, in step S195, a judgment is made to determine whether or not θ is equal to π. If it is not equal to π, processing advances to step S196, where θ is incremented by π/n. Then, processing returns to step S192, and the subsequent processing is executed again. When the judgment is made in step S195 that θ is equal to π, processing is terminated.--

Please amend the paragraph staring on page 38, line 2 to read as follows:

--For example, if the angular resolution is π/n, this processing enables a Hough transformation to be performed on a 64×64 pixel image by using at most 6n polygons. Also, if the angular resolution is π/n, a Hough transformation can be performed on 256×256 pixel image data by using at most 8n polygons.--